ISSN 1746-7659, England, UK
Journal of Information and Computing Science
Vol. 6, No. 4, 2011, *pp*. 243-254

A Genetic Algorithm Approach for Reliability of Bridge Network in Fuzzy System

G.S. Mahapatra[+] , P. Roy

Department of Engineering Sciences and Humanities Siliguri Institute of Technology, P.O.- Sukna, Siliguri-734009, West Bengal, India

(*Received August 30, 2011, accepted September 12, 2011*)

Abstract. This paper presents a Genetic Algorithm (GA) approach for solving constrained reliability optimization problem of the five unit bridge network. Considering uncertainty for cost, the reliability optimization problem has been solved by GA technique to maximize system reliability. GA is an efficient method for solving this type of optimization problems. This paper successfully applies the GA technique to obtain the optimal solution of the complex system reliability model under cost constraint in fuzzy environment in which the system cost and component costs might be imprecise. Triangular fuzzy number is used to represent the fuzzy cost coefficients. Static penalty method has been used to handle the cost constraint of the problem. To solve the reliability optimization model, we have developed the GA using MATLAB with tournament selection process, elitism mechanism, arithmetic crossover and uniform mutation operations. Finally, computational results are presented for the reliability of bridge network in crisp and fuzzy environment.

Keywords: Genetic algorithm, Reliability optimization, Penalty method, Bridge network, Fuzzy number

1. Introduction

Genetic Algorithm (GA) is a robust evolutionary optimization search technique that mimics the process of natural evolution to solve optimization problems. Genetic algorithms are based on the combination of principles of genetics and biological evolution. It is one of the heuristic optimization techniques, which include evolutionary strategies [1]. A genetic algorithm is a stochastic global optimization search method that uses natural selection in biological evolution as its model of problem solving as introduced by Holland [2] and further described by Goldberg [3]. John Holland [2][4] is the pioneer of genetic algorithm. GAs were invented and developed by him, his students and colleagues.

Several types of genetic algorithms have been developed to solve single or multiple objective optimization problems with any combination of linear/nonlinear objectives or constraints. Deng et al. [8] proposed the shredding genetic algorithm (SGA) that follows the mechanism adopted in modern breeding technology where healthy animals are cultivated by interfering with the natural selection process and filtering out pubs with undesirable characteristics using the principle of elitism. By simulating this filtration process, SGA can focus the search around the most important genes. Wang and Ghosn [9] proposed an extension algorithm of the SGA by combining the benefits of the shredding and learning operators to the linkage process proposed by Harik and Goldberg [10]. Cheng [11] proposed an artificial neural network based GA. Parallel GA technology [12] can be used to accelerate the speed of GAs. Bi-objective genetic algorithm [13] can be used for makespan and reliability optimized scheduling for workflow applications. Wang et al. [14] proposed the look-ahead genetic algorithm to intelligently optimize both the makespan and the reliability for a workflow application. Deb et al. [15] proposed the Non-dominated Sorting Genetic Algorithm (NSGA-II) which is a fast and very efficient multi-objective evolutionary algorithm. Chang et al. [17] proposed a two-phase sub population genetic algorithm to solve the parallel machine-scheduling problems. Chen et al. [25] proposed hybrid evolutionary algorithms with integration of GA and extremal optimization to solve the hot strip mill scheduling problem. Elsayed et al. [34] developed a multi-operator

[+] Corresponding author. Tel.: +919433135327; fax: +913532778003
E-mail address: g_s_mahapatra@yahoo.com

genetic algorithm and a self-adaptive multi-operator genetic algorithm for solving constrained optimization problems. Gonçalves and Resende [5] proposed a multi-population biased random-key genetic algorithm for the single container loading problem. Meng and Weng [7] proposed a GA to assessing work zone casualty risk.

GAs can be used to solve different type of optimization problems such as fuzzy, combinatorial and multi objective optimization problems. In last two decades, different optimization approaches have been proposed to solve reliability optimization problems. These include dynamic programming method, integer programming method, branch and bound method, Lagrange multiplier method, stochastic programming method, etc. ([20]-[24]). Today GA is widely used in reliability engineering to optimize reliability of a system under certain constraints. Painton and Campbell [26] proposed a GA to find maximum reliability to satisfy specific cost constraints. Coit and Smith [28] described a GA to solve the reliability optimization problem for a series-parallel system meeting the cost and weight constraints. Bhunia et al. [29] proposed a GA to solve reliability optimization problem of a series system with interval valued component reliability. Sahoo et al. [16] developed a GA to solve reliability redundancy optimization problem of series-parallel/parallel-series/complex system with interval valued reliability of each component. Cheng and Li [30] proposed a method for structural reliability analysis by integrating the uniform design method [31] with artificial neural network based GA. Hsieh et al. [32] utilized GAs to solve various types of reliability design problems, such as reliability optimization of series system, series-parallel system and complex systems. Yeh et al. [33] proposed a GA to solve the k-node set reliability optimization problem with capacity constraint of a distributed system. Gen and Cheng [6] described the applications of GAs to reliability optimization problems. Moghaddam et al. [35] proposed a GA for a redundancy allocation problem to the series-parallel system when the redundancy strategy can be chosen for individual subsystems. Kumar et al. [27] performed the reliability analysis of waste clean-up manipulator using real coded GA and fuzzy methodology. Kishor et al. [38] proposed a multi-objective GA for complex system reliability optimization problem under fuzzy environment, while minimizing the cost of the system.

In this paper, we have developed GA to solve the reliability optimization problem of a bridge network in crisp and fuzzy environment. Here, we have presented optimal solution of system reliability of bridge network with system cost as constraint in fuzzy and crisp environment. We have used real-value encoding to represent the solution as a chromosome. The initial population of size 40 has been generated randomly. Tournament selection process has been used to select the parent chromosomes from the population. Elitism process, arithmetic crossover operation and uniform mutation operation are applied to generate the offsprings that have higher fitness value than their parent. When a termination condition is satisfied, the GA will give the optimal solution of the problem. Finally, numerical exposure of the proposed GA technique is presented and some conclusion and recommendation are made.

2. Model Formulation

2.1. Notations

A bridge network reliability model is developed using GA technique under the following notations.

Notation	Definition
r_i	Reliability for i^{th} component of the reliability model
c_i	Cost coefficient for i^{th} component of the reliability model
$\tilde{c}_i$	Fuzzy cost coefficient for i^{th} component of the reliability model
a_i	Shape parameter for i^{th} component of the reliability model
C	Available system cost of the reliability model
$\tilde{C}$	Fuzzy system cost of the reliability model
c_i^l	Lower end point of the fuzzy cost interval for i^{th} component
c_i^u	Upper end point of the fuzzy cost interval for i^{th} component
C^l	Lower end point of the fuzzy system cost interval

C^u	Upper end point of the fuzzy system cost interval
$R_s(r_1,r_2,r_3,r_4,r_5)$	System reliability function of the reliability model
$C_s(r_1,r_2,r_3,r_4,r_5)$	System cost function of the reliability model
$\widetilde{C}_s(r_1,r_2,r_3,r_4,r_5)$	Fuzzy system cost function of the reliability model

2.2. Crisp model of bridge network

The bridge network is considered as a system of five components to find out the system reliability as shown in Figure 1. The algebraic expression for system reliability (R_s) of the bridge system is given as follows:

$$R_s(r_1,r_2,r_3,r_4,r_5) = r_5(r_1 + r_2 - r_1r_2)(r_3 + r_4 - r_3r_4) + (1 - r_5)(r_1r_3 + r_2r_4 - r_1r_2r_3r_4)$$

where $0 < r_i \le 1$ for $i = 1,2,3,4,5$.

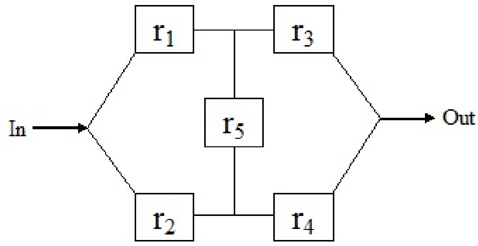

Fig. 1: Schematic of five unit bridge network

The cost constraint for the bridge system is given as follows:

$$C_s(r_1,r_2,r_3,r_4,r_5) = \sum_{i=1}^{5} c_i r_i^{a_i} \le C$$

Hence, mathematically, the reliability optimization problem for the bridge network can be formulated as follows:

$$\begin{aligned} &\textit{Maximize } R_s(r_1,r_2,r_3,r_4,r_5) = r_5(r_1 + r_2 - r_1r_2)(r_3 + r_4 - r_3r_4) \\ &\qquad\qquad + (1 - r_5)(r_1r_3 + r_2r_4 - r_1r_2r_3r_4) \\ &\textit{subject to } C_s(r_1,r_2,r_3,r_4,r_5) = \sum_{i=1}^{5} c_i r_i^{a_i} \le C \end{aligned} \tag{1}$$

where $0 < r_i \le 1$ for $i = 1,2,3,4,5$.

2.3. Fuzzy bridge network model

In real life, the values of cost coefficient of components and the available system cost are not rigid. These costs are not known with certainty. We need to consider uncertainty for cost. So considering uncertainty for the cost coefficients and system cost, it is more realistic to consider them as fuzzy.

In fuzzy system, the cost constraint for the bridge network is given as follows:

$$\widetilde{C}_s(r_1,r_2,r_3,r_4,r_5) = \sum_{i=1}^{5} \widetilde{c}_i r_i^{a_i} \le \widetilde{C}$$

Here fuzzy cost coefficients ($\widetilde{c}_i$) and fuzzy system cost ($\widetilde{C}$) are represented by triangular fuzzy number as follows:

$$\widetilde{c}_i = (c_i^l, c_i, c_i^u) \text{ for } i = 1,2,3,4,5, \text{ and } \widetilde{C} = (C^l, C, C^u)$$

Applying α - cut operation on the fuzzy numbers, we get the crisp interval of costs as follows:

$$c_i(\alpha) = [c_i^l + \alpha(c_i - c_i^l), c_i^u - \alpha(c_i^u - c_i)] \text{ for } i = 1,2,3,4,5,$$

$$\text{and } C(\alpha) = [C^l + \alpha(C - C^l), C^u - \alpha(C^u - C)]$$

Hence, the reliability optimization problem for the bridge system in fuzzy environment can be formulated as follows:

$$\text{Maximize } R_s(r_1, r_2, r_3, r_4, r_5) = r_5(r_1 + r_2 - r_1 r_2)(r_3 + r_4 - r_3 r_4) + (1 - r_5)(r_1 r_3 + r_2 r_4 - r_1 r_2 r_3 r_4)$$

subject to

$$\sum_{i=1}^{5} [c_i^l + \alpha(c_i - c_i^l), c_i^u - \alpha(c_i^u - c_i)] r_i^{a_i} \le [C^l + \alpha(C - C^l), C^u - \alpha(C^u - C)]$$

where $0 < r_i \le 1$ for $i = 1,2,3,4,5$ and $0 \le \alpha \le 1$.

The above constraint can be written as follows:

$$\sum_{i=1}^{5} (c_i^l + \alpha(c_i - c_i^l)) r_i^{a_i} \ge C^l + \alpha(C - C^l) \tag{2}$$

and

$$\sum_{i=1}^{5} (c_i^u - \alpha(c_i^u - c_i)) r_i^{a_i} \le C^u - \alpha(C^u - C) \tag{3}$$

Therefore the bridge network reliability model (1) transform into the following form (4):

$$\begin{aligned} \textit{Maximize } R_s(r_1, r_2, r_3, r_4, r_5) &= r_5(r_1 + r_2 - r_1 r_2)(r_3 + r_4 - r_3 r_4) \\ &\quad + (1 - r_5)(r_1 r_3 + r_2 r_4 - r_1 r_2 r_3 r_4) \end{aligned} \tag{4}$$

subject to

$$\sum_{i=1}^{5} (c_i^l + \alpha(c_i - c_i^l)) r_i^{a_i} \ge C^l + \alpha(C - C^l)$$

$$\sum_{i=1}^{5} (c_i^u - \alpha(c_i^u - c_i)) r_i^{a_i} \le C^u - \alpha(C^u - C)$$

where $0 < r_i \le 1$ for $i = 1,2,3,4,5$ and $0 \le \alpha \le 1$.

3. GA Based Constraint Handling Approach

Here we have discussed GA based constraint handling approach of the optimization problem. The penalty method has been used to handle the constraint of the optimization problem. We have used static penalty [36] to transform the constrained optimization problem to an unconstrained optimization problem.

Let us consider the following constrained optimization problem:

$$\text{Maximize } f(x)$$

subject to

$$h_i(x) \le 0, \quad i = 1, 2, ..., m$$

where m is the number of constraints.

According to the static penalty function method, this problem can be transformed to an unconstrained maximization problem and the objective function can be written as follows:

$$\text{Maximize } F(x) = f(x) - \sum_{i=1}^{m} \sigma_i P(x)$$

where $P(x) = \max[0, h_i(x)]^2$.

In the above equation, P(x) is the penalty function which represents the square of constraint violation and σ_i is the penalty coefficient or penalty parameter for ith constraint.

$$F(x)=\begin{cases} f(x) & if \;\; x \in S \\ f(x)-\sum_{i=1}^{m}\sigma_i P(x) & if \;\; x \notin S \end{cases}$$

where S is the set of feasible solutions.

If no constraint violation occurs i.e. the value of x is a feasible solution, P(x) will be zero for all constraints. If constraint violation occurs for some constraint, P(x) will be positive for that violated constraint.

Here the penalty parameters (σ_i) do not depend on the current generation number and a constant penalty is applied to unfeasible solutions.

4. GA Implementation

A GA has been developed to solve the reliability optimization problem. The steps of the proposed GA are given as follows:

Algorithm

Step 1: Initialize the parameters of the genetic algorithm as follows: population size=40, crossover rate=0.9, mutation rate=0.05, maximum number of generations=300.

Step 2: Initialize different parameters of the optimization problem.

Step 3: Set g=1, where g signifies the current generation number.

Step 4: Initialize the chromosomes of the population P(g) [P(g) represents the population of gth *generation] taking the lower and upper bounds of the variables as 0 and 1 respectively.*

Step 5: Evaluate the fitness value of each chromosome in the population P(g) considering the fitness function.

Step 6: Select parent chromosomes from P(g) by tournament selection process.

Step 7: Alter P(g) by applying elitism process, arithmetic crossover and uniform mutation operations on the selected parent chromosomes.

Step 8: Evaluate the fitness value of each chromosome (offspring) in the population P(g).

Step 9: Find the chromosome with best fitness value from P(g).

Step 10: If the termination condition is satisfied, go to step 14.

Step 11: Place P(g) in P(g+1).

Step 12: Increase g by unity.

Step 13: Go to step 5.

Step 14: Print the chromosome of P(g) with the best fitness function value (which is the solution of the optimization problem).

Step 15: End.

The flowchart of the GA is shown in Figure 2.

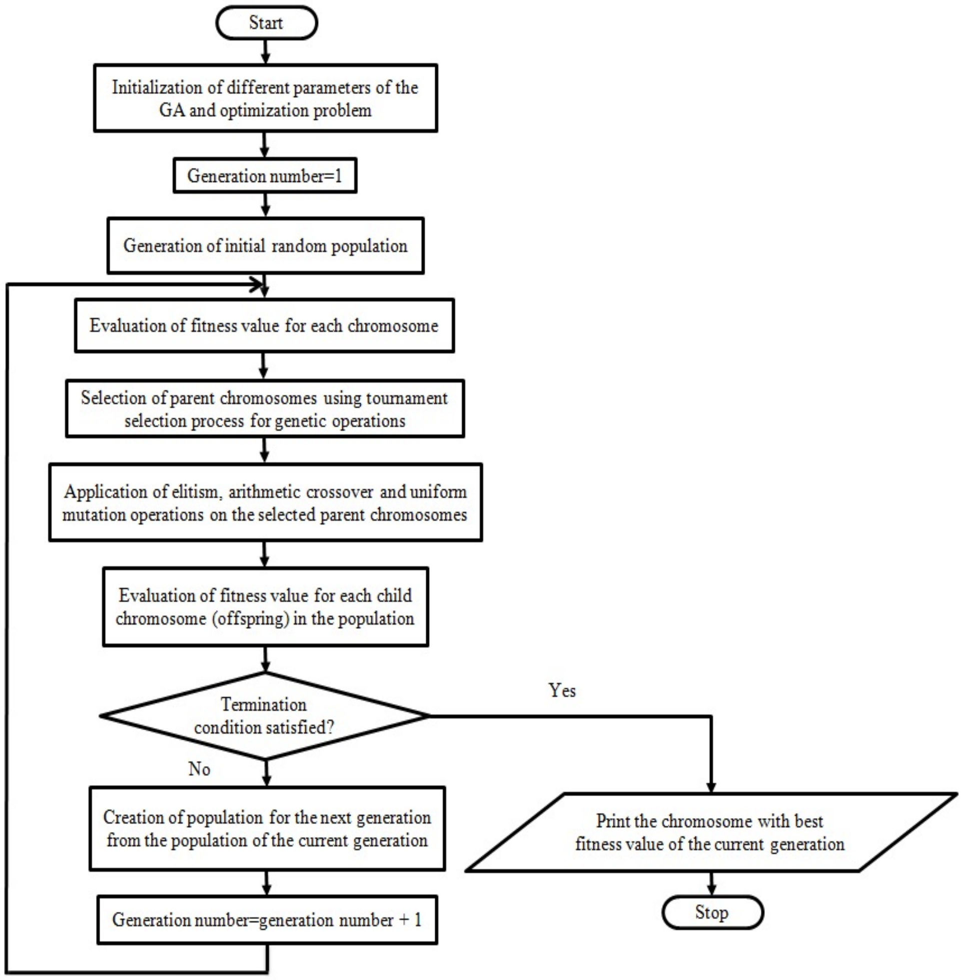

Fig. 2: Flowchart of the proposed GA for optimize system reliability

4.1. Developed GA for reliability optimization of crisp model of bridge system

The GA developed for crisp model of bridge network reliability optimization problem is described as follows:

Chromosomal-encoding of solution. In GA, each solution is represented by a chromosome. A chromosome is an encoded solution. In this problem, the length of a chromosome i.e. the number of genes in a chromosome is 5 as there are five components in the model. Real value encoding has been used to encode the solution as a chromosome and each component (gene) of the chromosome represents the reliability of the component of the model. The chromosome representation of a solution is given below as an example:

$$[0.78\ 0.88\ 0.91\ 0.70\ 0.93]$$

Initial population

A set of possible solutions i.e. chromosomes in a generation is called a population. The GA is started with an initial population. The initial population has been determined randomly with a population size of 40.

Fitness function

The fitness function is defined as a combination of the objective function and a static penalty function. The fitness function is used to determine the fitness value of each chromosome in a generation. The fitness value indicates which chromosomes will be selected as parent to be involved in genetic operations.

The fitness function of the bridge system reliability optimization problem is given below:

$$\text{Maximize } R_s - \sigma.\max[0, C_s - C]^2$$

i.e.

$$\begin{aligned}\text{Maximize}\, & r_5(r_1+r_2-r_1r_2)(r_3+r_4-r_3r_4)+(1-r_5)(r_1r_3+r_2r_4-r_1r_2r_3r_4) \\ & -\sigma.\max[0, c_1r_1^{a_1}+c_2r_2^{a_2}+c_3r_3^{a_3}+c_4r_4^{a_4}+c_5r_5^{\alpha_5}-C]^2\end{aligned} \tag{5}$$

Selection procedure

The selection process selects chromosomes according to their fitness value for genetic operations. Chromosomes are selected from the population as parents. To form new offspring, chromosomes are selected according to the Darwin's theory about evolution "Survival of the fittest" because the more suitable they are, the more chances to reproduce. Selection is performed to emphasize fitter individuals in the population in hope that their offsprings have higher fitness value. Here, tournament selection process of size three has been used for chromosome selection.

Elitism

Elitism method copies the best chromosome or the few best chromosomes of the current generation to the next generation without crossover and mutation. The rest of the individuals are passed through the crossover and mutation operations. The best chromosome may be lost in any generation by crossover and mutation operations when a new population is created. To overcome this situation elitism method has been applied. Here, the value of elite count has been taken as one. Elite count indicates the number of individuals that will be guaranteed to survive to the next generation without crossover and mutation operations.

Genetic crossover operator

After selection process, the crossover operator is applied to the resulting parent chromosomes. The main purpose of crossover is to exchange genetic information between randomly selected parent chromosomes. Crossover recombines the genetic material of two or more parent chromosomes to produce new individuals (offsprings) for the next generation. We have used the arithmetic crossover operator [37] which defines a linear combination of two chromosomes. Suppose C_1^g and C_2^g are two parent chromosomes that are randomly selected for crossover in g^{th} generation. If C_1^g and C_2^g are to be crossed, the resulting offspring are C_1^{g+1} and C_2^{g+1}, which is a linear combination of their parents, i.e.

$$C_1^{g+1} = \beta C_1^g + (1-\beta)C_2^g$$
$$C_2^{g+1} = (1-\beta)C_1^g + \beta C_2^g$$

where β is a random value that lies between 0 and 1.

Here, the arithmetic crossover operator has been applied based on the crossover probability of 0.9.

Genetic mutation operator

Mutation operation is performed to introduce random variations into the genes of a chromosome. Mutation is applied to prevent falling of all solutions in population into a local optimum of the solved problem. So this operation is performed to improve the global optimal solution. Mutation helps to regain the information lost in earlier generations. Mutation operation is applied to a single chromosome with a very low rate. Mutation can change single or all the genes of a randomly selected chromosome. In this problem, we have used the uniform mutation operator with a mutation rate of 0.05. It selects a fraction of the vector entries of an individual for mutation, where each entry has the same probability as the mutation rate of being mutated.

Terminating criteria

The termination condition is used to stop the GA. The GA is terminated when any one of the following conditions is satisfied:

(i) the number of generations has been reached to the maximum number of generations.

(ii) the specified time has been elapsed.

(iii) the average change in the fitness function value over specified generations is less than a very small pre-assigned positive number.

Here, we have used 300 generations as the maximum number of generations for the GA, 100 generations as the specified generations and 10^{-50} as a very small positive number.

Parameters of the GA

There are four parameters used in the GA that affect the solution obtained from the GA. These parameters are population size (ps), crossover rate/probability (cr), mutation rate/probability (mr) and maximum number of generations (m_gen). Here, we have taken the value of GA parameters as follows:

ps=40, cr=0.9, mr=0.05 and m_gen=300.

4.2. Developed GA for bridge network reliability in fuzzy environment

The GA developed for fuzzy bridge network reliability optimization problem is described here.

Chromosomal-encoding of solution.

In fuzzy environment, when we consider α as a variable, the chromosome size will be 6. When α is not a variable i.e. we give the value of α in the optimization problem, then the chromosome size is 5.

Initial population

The initial population has been generated randomly with population size of 40 taking the lower and upper bounds of the variables as 0.7 and 0.99 respectively.

Fitness function

The fitness function of the fuzzy bridge network reliability optimization model is given below.

$$\begin{aligned}
&\text{Maximize } r_5(r_1+r_2-r_1r_2)(r_3+r_4-r_3r_4)+(1-r_5)(r_1r_3+r_2r_4-r_1r_2r_3r_4)\\
&-\sigma_1 \max[0,-(c_1^l+(c_1-c_1^l)\alpha)r_1^{a_1}-(c_2^l+(c_2-c_2^l)\alpha)r_2^{a_2}-(c_3^l+(c_3-c_3^l)\alpha)r_3^{a_3}\\
&-(c_4^l+(c_4-c_4^l)\alpha)r_4^{a4}-(c_5^l+(c_5-c_5^l)\alpha)r_5^{a_5}+(C^l+(C-C^l)\alpha)]^2\\
&-\sigma_2 \max[0,(c_1^u-(c_1^u-c_1)\alpha)r_1^{a_1}+(c_2^u-(c_2^u-c_2)\alpha)r_2^{a_2}+(c_3^u-(c_3^u-c_3)\alpha)r_3^{a_3}\\
&+(c_4^u-(c_4^u-c_4)\alpha)r_4^{a_4}+(c_5^u-(c_5^u-c_5)\alpha)r_5^{a_5}-(C^u-(C^u-C)\alpha)]^2
\end{aligned} \tag{6}$$

The selection procedure, elitism mechanism, crossover operator, mutation operator, terminating criteria and parameters of the GA developed for fuzzy bridge system reliability optimization problem are same with the GA developed for crisp model of bridge system.

5. Numerical Presentation

5.1. Solution of bridge network reliability in crisp model

The proposed GA has been implemented using MATLAB.

Let the cost coefficients (c_i), shape parameters (a_i) and the available system cost (C) used for (1) are given as follows:

$c_1=24,\ c_2=20,\ c_3=18,\ c_4=16,\ c_5=15; a_1=0.68,\ a_2=0.73,\ a_3=0.71,\ a_4=0.67,\ a_5=0.70,$ and $C=85$.

Here, we have taken the value of penalty parameter (σ) as 999.

Because of the stochastic nature of GA, 20 trials have been performed for 300 generations as the maximum number of generations each and the best solution from among the 20 trials has been considered as the final optimal solution.

In the Figure 3, plot of the generation number versus the best fitness value in those generations is presented. The figure shows the best fitness value in the specified generations for the trial for which we get the best solution. From the graph, we can see that as the generation number increases, the best fitness value in those generations converge to the optimal solution.

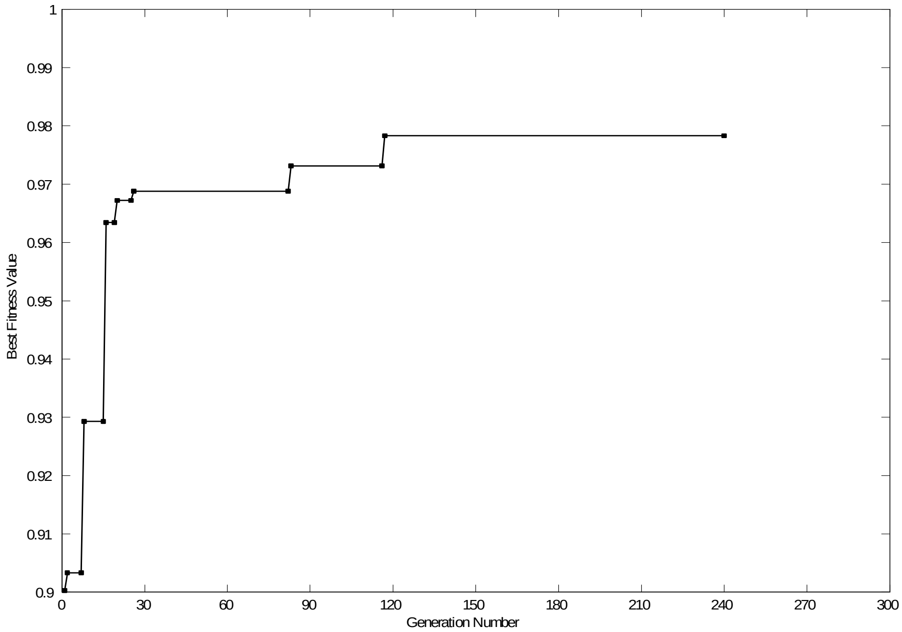

Fig. 3: Fitness value convergence

After 240 generations (of the trial for which we get the best solution), we get the following solutions as the optimal solution identified by the GA:

The maximum system reliability of bridge network is $R_s = 0.9783$.

The maximum reliability of the five components are as follows:

$$r_1 = 0.929,\ r_2 = 0.832,\ r_3 = 0.862,\ r_4 = 0.962,\ r_5 = 0.803.$$

And the system cost of bridge network is $C_s = 84.9682$.

Here, the cost constraint has been satisfied efficiently for the chosen value of the penalty parameter and we get the maximum system reliability satisfying the cost constraint.

The chosen value of the penalty parameter for our reliability optimization problem efficiently handles the cost constraint of the problem.

5.2. Solution of fuzzy bridge network

The GA has been implemented with MATLAB. Let the fuzzy cost coefficients $(\tilde{c}_i)$ of five components of the bridge network and the fuzzy system cost $(\tilde{C})$ of the bridge model are given as follows:

$\tilde{c}_1 = (22.8, 24, 25)$, $\tilde{c}_2 = (19.5, 20, 21.3)$, $\tilde{c}_3 = (17, 18, 19.2)$, $\tilde{c}_4 = (15.3, 16, 17)$, $\tilde{c}_5 = (14, 15, 16.1)$ and $\tilde{C} = (80, 85, 94)$.

Applying $\alpha -$cut operation on the above fuzzy numbers, we get the crisp interval of costs as follows:

$$c_1(\alpha) = [22.8 + 1.2\alpha, 25 - \alpha], c_2(\alpha) = [19.5 + 0.5\alpha, 21.3 - 1.3\alpha],$$

$$c_3(\alpha) = [17 + \alpha, 19.2 - 1.2\alpha],\ c_4(\alpha) = [15.3 + 0.7\alpha, 17 - \alpha],$$

$$c_5(\alpha) = [14 + \alpha, 16.1 - 1.1\alpha] \text{ and } C(\alpha) = [80 + 5\alpha, 94 - 9\alpha]$$

In this optimization problem, two penalty parameters σ_1 and σ_2 are used for the two constraints and we have taken the value of σ_1 and σ_2 as 10^9.

Twenty trials have been performed for the proposed GA and the best solution from among the 20 trials has been taken as the final optimal solution.

After 141 generations (of the trial for which we get the best solution), we get the following solutions as the optimal solution identified by the proposed GA:

$$\alpha = 0.81,\ r_1 = 0.946,\ r_2 = 0.85,\ r_3 = 0.966,\ r_4 = 0.839,\ r_5 = 0.817 \text{ and } R_s = 0.9845.$$

Now in the second part of this subsection, we have presented a solution approach of the said model for different values of α.

In Table 1, we have shown the values of system reliability (R_s) and reliability of the five components for different values of α. We have also mentioned the generation number after which we get the reliability of the five components and system reliability as the optimal solution identified by the GA.

Table 1: Optimum reliability for different values of α

	$\alpha = 0$	$\alpha = 0.25$	$\alpha = 0.5$	$\alpha = 0.75$	$\alpha = 0.99$
r_1	0.881	0.896	0.889	0.875	0.965
r_2	0.878	0.894	0.898	0.897	0.826
r_3	0.898	0.898	0.897	0.899	0.78
r_4	0.893	0.896	0.891	0.895	0.979
r_5	0.894	0.897	0.887	0.896	0.847
R_s	0.9726	0.9768	0.9756	0.9747	0.9837
Generation Number	201	300	293	219	106

The constraints have been satisfied efficiently for the chosen value of penalty parameters and we get the most promising optimal solutions for that value of penalty parameters.

6. Conclusion

In this paper, the bridge network reliability optimization problem in crisp and fuzzy environment has been solved by GA approach. Here, the costs of the model have been considered as fuzzy because of uncertainty. For handling constraints, static penalty method has been utilized to solve the constrained optimization problem. We have developed a GA to solve the reliability optimization problem with tournament selection process, arithmetic crossover, uniform mutation and elitism of size one. We have find out the system reliability and reliability of different components of the model using the proposed GA. Our numerical results show that the GA developed for this reliability optimization problem give very promising results as the optimal solution for the chosen values of penalty parameter. The demonstrations show that the GA approach can be a very powerful tool for solving reliability optimization problems of bridge network.

7. Acknowledgements

The authors would like to acknowledge the Council of Scientific and Industrial Research (CSIR), New Delhi, India, for providing financial support to carry out this research work.

8. References

[1] C.R. Reeves. *Modern Heuristic Techniques for Combinatorial Problems*. John Wiley & Sons, 1993.

[2] J.H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, 1975.

[3] D.E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison Wesley, 1989.

[4] J.H. Holland. Genetic algorithm and the optimal allocation of trials. *SIAM J. Comput*. 1973, **2**(2): 88-105.

[5] J.F. Gonçalves and M.G.C. Resende. A parallel multi-population biased random-key genetic algorithm for a container loading problem. *Comput. Oper. Res*. 2011, **39**(2): 179-190.

[6] M. Gen and R. Cheng. Genetic Algorithms and Engineering Design. John Wiley & Sons, 1997.

[7] Q. Meng and J. Weng. A Genetic algorithm approach to assessing work zone casualty risk. *Safety Sci*. 2011, **49**(8-9): 1283-1288.

[8] L. Deng, M. Ghosn and S. Shao. Shredding genetic algorithm for reliability analysis of structural systems. *J. Struct. Safety*. 2005, **27**(2): 113-131.

[9] J. Wang and M. Ghosn. Linkage-shredding genetic algorithm for reliability assessment of structural systems. *J. Struct. Safety*. 2005, **27**(1): 49-72.

[10] G.R. Harik and D.E. Goldberg. Linkage learning through probabilistic expression. *J. Comput. Meth. Appl. Mech. Eng*. 2000, **186** (2-4) : 295-310.

[11] J. Cheng. Hybrid genetic algorithms for structural reliability analysis. *J. Comput. Struct*. 2007, **85**(19-20): 1524-1533.

[12] D. Lim, Y. S. Ong, Y. Jin, B. Sendhoff and B. S. Lee. Efficient hierarchical parallel genetic algorithms using grid computing. *Future Gener. Comp. Sy*. 2007, **23**(4): 658-670.

[13] A. Doğan and F. Özgüner. *Biobjective scheduling algorithms for execution time-reliability trade-off in heterogeneous computing systems*. *Comput. J*. 2005, **48**(3): 300-314.

[14] X. Wang, C.S. Yeo, R. Buyya and J. Su. Optimizing the makespan and reliability for workflow applications with reputation and a look-ahead genetic algorithm. *Future Gener. Comp. Sy*. 2011, **27**(8): 1124-1134.

[15] K. Deb, S. Agarwal, A. Pratap and T. Meyarivan. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. *Proc. of the Parallel Problem Solving from Nature VI Conference*. 2000, pp. 849-858.

[16] L. Sahoo, A.K. Bhunia and D. Roy. A genetic algorithm based reliability redundancy optimization for interval valued reliabilities of components. *J. Appl. Quant. Meth*. 2010, 5(2): 270-287.

[17] P.C. Chang, S.H. Chen and K.L. Lin. Two-phase sub population genetic algorithm for parallel machine-scheduling problem. *Expert Syst. Appl*. 2005, **29**(3): 705-712.

[18] S.N. Sivanandam and S.N. Deepa. *Introduction to Genetic Algorithms*. Springer, 2008.

[19] M. Gen and R. Cheng. Genetic Algorithms and Engineering Optimization. John Wiley & Sons, 2000.

[20] F.A. Tillman, C.L. Hwang and W. Kuo. Optimization techniques for system reliability with redundancy - a review. *IEEE T. Reliab*.1977, **R-26**(3): 148-155.

[21] W. Kuo, V.R. Prasad, F.A. Tillman and C.L. Hwang. Optimal Reliability Design: Fundamentals and Applications. Cambridge University Press, 2001.

[22] X.L. Sun, K.I.M. Mckinnon and D. Li. A convexification method for a class of global optimization problems with applications to reliability optimization. *J. Global Optim*. 2001, **21**(2): 185-199.

[23] X.L. Sun and D. Li. Optimality condition and branch and bound algorithm for constrained redundancy optimization in series systems. *Optim. Eng*. 2002, **3**(1): 53-65.

[24] R. Zhao and B. Liu. Stochastic programming models for general redundancy- optimization problems. *IEEE T. Reliab*. 2003, **R52**(2): 181-191.

[25] Y.W. Chen, Y.Z. Lu, M. Ge, G.K. Yang and C.C. Pan. Development of hybrid evolutionary algorithms for production scheduling of hot strip mill. *Comput. Oper. Res*. 2011, **39**(2): 339-349.

[26] L. Painton and J. Campbell. Genetic algorithms in optimization of system reliability. *IEEE T. Reliab*. 1995, **44**(2): 172-178.

[27] N. Kumar, J.H. Borm and A. Kumar. Reliability analysis of waste clean-up manipulator using genetic algorithms and fuzzy methodology. *Comput. Oper. Res*. 2011, **39**(2): 310-319.

[28] D.W. Coit and A.E. Smith. Reliability optimization of series-parallel systems using a genetic algorithm. *IEEE T. Reliab*. 1996, **45**(2) (1996): 254-260.

[29] A.K. Bhunia, L. Sahoo and D. Roy. Reliability stochastic optimization for a series system with interval component reliability via genetic algorithm. *Appl. Math. Comput*. 2010, **216**(3): 929-939.

[30] J. Cheng and Q.S. Li. Reliability analysis of structures using artificial neural network based genetic algorithms. *Comput. Method Appl*. 2008, **M. 197**(45-48): 3742-3750.

[31] K.T. Fang and Y. Wang. Number-Theoretic Methods in Statistics. Chapman & Hall, 1994.

[32] Y.C. Hsieh, T.C. Chen and D.L. Bricker. Genetic algorithms for reliability design problems. Microelectron. Reliab. 1998, **38**(10): 1599-1605.

[33] Y.S. Yeh, C.C. Chiu and R.S. Chen. A genetic algorithm for k-node set reliability optimization with capacity constraint of a distributed system. *Proc. of the National Science Council*. 2001, pp. 27-34.

[34] S.M. Elsayed, R.A. Sarker and D.L. Essam. Multi-operator based evolutionary algorithms for solving constrained optimization problems. *Comput. Oper. Res*. 2011, **38**(12): 1877-1896.

[35] R. Tavakkoli-Moghaddam, J. Safari and F. Sassani. Reliability optimization of series-parallel systems with a choice of redundancy strategies using a genetic algorithm. *Reliab. Eng. Syst. Safe*. 2008, **93**(4): 550-556.

[36] A. Homaifar, S.H.Y. Lai and X. Qi. Constrained optimization via genetic algorithms. *Simulation*. 1994, **62**(4): 242-254.

[37] Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer, 1996.

[38] A. Kishor, S.P. Yadav and S. Kumar. A Multi-objective Genetic Algorithm for Reliability Optimization Problem. *Int. J. Perform. Eng*. 2009, **5**(3): 227-234.